

A Monitoring Infrastructure for FPGA Self-Awareness and Dynamic Adaptation

Carlos Gómez Osuna Miguel Ángel Sánchez Marcos Pablo Ituero Marisa López-Vallejo

Dpto. de Ingeniería Electrónica, ETSI Telecomunicación, Universidad Politécnica de Madrid, Spain

{carlosgo, masanchez, pituero, marisa}@die.upm.es

Abstract—Variabilities associated with CMOS evolution affect the yield and performance of current digital designs. FPGAs, which are widely used for fast prototyping and implementation of digital circuits, also suffer from these issues. Proactive approaches start to appear to achieve self-awareness and dynamic adaptation of these devices. To support these techniques we propose the employment of a multi-purpose sensor network. This infrastructure, through adequate use of configuration and automation tools, is able to obtain relevant data along the life cycle of an FPGA. This is realised at a very reduced cost, not only in terms of area or other limited resources, but also regarding the design effort required to define and deploy the measuring infrastructure. Our proposal has been validated by measuring inter-die and intra-die variability in different FPGA families.

Index Terms—Monitoring, On-Chip, FPGA, CMOS, temperature, variability, leakage, aging, dynamic adaptation.

I. INTRODUCTION

The amazing advancements that CMOS technology has undergone have come at the cost of certain drawbacks such as parameters uncertainties, unpredictable thermal behaviour or exacerbated wear-out and aging processes. Once chips are fabricated, current digital systems deal with these issues through self-awareness and dynamic adaptation.

FPGAs are nowadays the flagship of dedicated electronic circuits. Their large capacity allows the implementation of complex applications and their partial reconfiguration abilities make possible the dynamic adaptation to particular circumstances. In order to react to the changing environment and circuit conditions, in field measurement of significant magnitudes through the insertion of sensors and monitoring networks have become a promising solution [1]. Budget constraints make designers exploit the capacity of FPGAs as much as possible, thus, this monitoring infrastructure must suppose little overhead [2]. Furthermore, as FPGAs are often used with extremely short time-to-market demands, a methodology towards the automation to design and incorporate this infrastructure is crucial, albeit missing to date.

Current work on this field includes the usage of ring oscillators [3], launch and capture circuitry [1], configurable clock sources [4], or clocked delay chains [5] in order to measure temperature, aging, clock variability, etc. Almost none of these approaches addresses the concerns about how difficult it may be to introduce monitoring at different stages, and in

some cases, sensors make use of scarce resources (such as system clocks in [5] or DCMs in [4]).

Focusing on FPGA architectures, this paper copes with the problem of environmental and intrinsic uncertainties by means of a light-weight monitoring network of delay sensors. These sensors can monitor process, critical path, temperature and aging variations. The network can be employed to obtain characterisation maps of the device such as process- or aging-dependent delay maps, and it can also be used for online monitoring of these magnitudes along with temperature, thus providing support for partial reconfiguration. We have employed reusable and configurable structures, allowing a high degree of automation, and lowering the effort to re-deploy the network in different configurations during a device life cycle. To illustrate the usefulness of the design, we first employ this network to obtain a measure of inter-die variations in a batch of 36 identical FPGAs; finally, targeting a device with a very large number of logic elements, we construct an intra-die variability map.

The paper is organised as follows. Section II describes the proposed infrastructure and the procedure to incorporate it. Inter- and intra- die variations case studies are developed and analysed in sections III and IV, respectively. Finally some conclusions and future lines are drawn in section V.

II. SENSING INFRASTRUCTURE

A. Sensor and Sensor Network

We have employed delay-based measurements to obtain performance and operation point parameters of the devices under study. Delay sensors have already been proposed in the literature as a way to measure the temperature and several types of variations [1], [3], [6] because logic and interconnect delay present a well-known dependency on the temperature, Vdd voltage, aging, etc. These sensors rely on a pulse generator to feed a signal into a delay chain and include a loop counter that controls the number of pulses being fed into the chain. The longer the delay of the sensor, the higher the accuracy at the expense of an incremented area; once the range and accuracy requirements of the expected measurements are known, a valid delay must be chosen.

As explained, these sensors can be reused to obtain estimates of various operational and performance parameters, depending on how they are configured and deployed. Delay sensors are encapsulated in such a way that can be easily adapted and placed by the modification of generic parameters:

This work was funded by the CICYT project DELTA TEC2009-08589 of the Spanish Ministry of Economy and Competitiveness and the Dpto. de Ingeniería Electrónica of the Universidad Politécnica de Madrid.

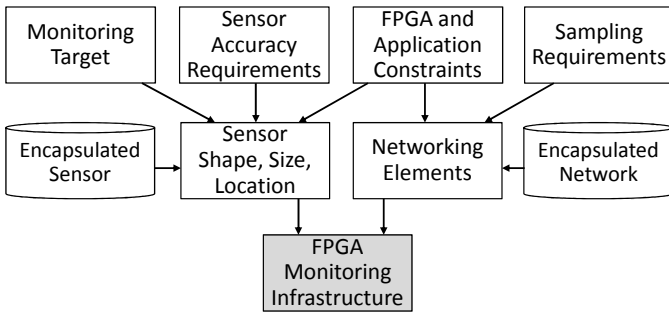


Fig. 1. Infrastructure Implementation Overview.

- Location. Each sensor includes (x, y) coordinates for specific placement.
- Size. Both in terms of the length of the delay chain and in the total delay introduced by the sensor.
- Shape. The sensor can be shaped to fit into unused logic or to mimic the length and layout of a critical path.

As sensor interface, we have implemented a shared time-to-digital converter, making use of the features of the lightweight monitoring network proposed in [2]. This network is composed by one front-end node, which centralises the time to digital converter and serves as an interface with host systems, and a back-end node for each sensor. Our implementation of the network allows for quick adaptation to different number of sensors. It also permits the inclusion in the front-end of different quantization capabilities and a calibration unit. Since all sensors in a network share the same single-wire data line to indicate the end of their activity to the front-end, a measurement error (in sampling clock cycles) is introduced which is linear with the number of nodes in the network, i.e. $\epsilon \leq N_{nodes} + k$ cycles, where N_{nodes} is the number of nodes in the network, and k is an implementation dependent constant accounting for control overhead, 4 in our case. The designer must select the sampling clock frequency, so that the error introduced by the network will not have a substantial effect on the measurements.

B. Infrastructure Implementation Procedure

This section describes our approach towards building an FPGA monitoring infrastructure (figure 1). The infrastructure is composed of a collection of sensors connected by a set of networking elements.

In order to establish the shape, size and location of the sensors we need the following information: (a) The monitoring target (e.g. device characterisation, online measurements, etc.); (b) sensing accuracy requirements; (c) timing and resource limitations imposed by the device and the application; (d) the encapsulated sensor description.

The accuracy of the actual measurements will be affected not only by the sensor, but also by the sampling frequency and the number of nodes in the network as explained previously. Therefore to define the networking elements (number and size of networks), the next inputs are required: (a) The FPGA

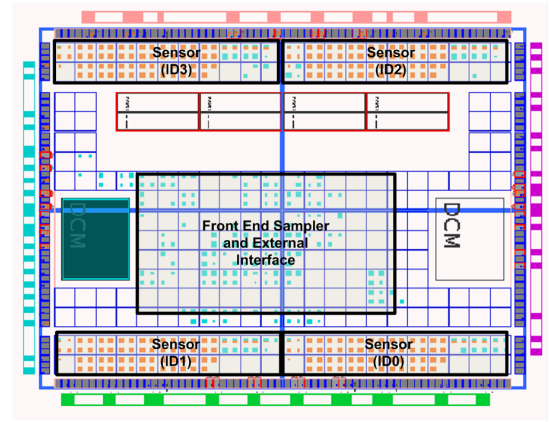


Fig. 2. Spartan 3E100. Four-Sensor Network Placement.

and application restrictions including resources usage, floor-planning and timing; (b) the sampling requirements including sampling frequency and maximum tolerated error; (c) an encapsulated network architecture.

A degree of automation is achieved through the usage of template files for the encapsulated sensors and networks, and definition of application-specific parameters in configuration files (size, shape and location for each sensor and network error tolerance). These files are employed to automatically generate the RTL code defining the monitoring infrastructure.

III. INTER-DIE VARIATIONS

In our first tests with the proposed sensing architecture we estimated inter-die variability in a group of 36 low cost 90nm Spartan-3E 100-4 devices mounted in Digilent's Basys 2 training boards. For this device, the main timing limitations are the minimum clock pulse width (T_{ch} , T_{cl} , 0.80ns), LUT delay (T_{ilo} 0.76ns) and DFF clock-to-output T_{cko} , 0.60ns [7].

A. Test Setup

In order to measure inter-die variations, a network of 4 sensors was placed in each FPGA (depicted in Fig. 2). Four sensors were used in each device in order to avoid measurement glitches and to illustrate the usefulness of the networking infrastructure. The exact features of each sensor were obtained as described next.

First, it is necessary to identify the required accuracy for the measurements, and thus fix the delay of the sensors, but always with regard to the exact timing characteristics of the FPGA, ensuring that sensors do not violate any basic conditions outlined in the device datasheet in terms of clock pulse width and setup time. According to [8], σ variation of ring oscillators for a 90nm process is on the range 3.8% to 7.5%. For our measurements, the theoretical delay of a sensor, as defined by adding up all delay elements on its chain and accounting for the upper limit of the delay loop counter, was chosen to be $140\mu s$ as an appropriate value.

The next step is to select the number of delay stages and the size of the loop counter to obtain the necessary base delay. We define a delay-stage as a fully utilised slice, i.e.

TABLE I
SYNTHESIS, THEORETICAL, PPR AND MEASURED RESULTS FOR DIFFERENT DELAY SENSORS.

Spartan 3E-S100E						
Delay Stages	Loop Counter	Theoretical Delay (ns)	Post-PAR Delay (ns)	Measured Delay (ns)	Total Area (slices)	Delay Area (%)
10	2,560	139,264	189,250	124,170	34	29%
20	1,280	139,264	177,330	117,500	43	47%
30	853	139,210	176,250	116,340	50	60%
Virtex 5 LX50T						
4	4,096	117,964.8	163,030	92,370	14	29%

a slice where all LUT+Latch pairs have been employed to add delay to the chain. Table I summarises synthesis results for different delay-chain and loop counter size combinations, along with basic data on the theoretical expected measurement (according to the datasheet), post-Place and Route simulated measurement (according to manufacturer worst case models) and actual measurements from one of the devices. Very similar results can be achieved with various combinations of delay and count, although care must be taken that the size of the counter does not outgrow the delay generator itself. Note that, in terms of area, a much smaller sensor could be achieved if necessary through the usage of counters based on shift-register and the Chinese Remainder Theorem [3].

The sampling frequency can be chosen at will because no application logic exists during device characterisation. With a sampling frequency of 100MHz each sensor will yield a total count of 14000 cycles, with a maximum 3.7% to 7.5% deviation of 518 - 1050 cycles and a maximum network-injected error of 9 cycles per measurement, or about 1.7% of the minimum expected variation.

There are significant differences between the expected delay, and those obtained through simulation and through actual measurement. The higher delay obtained during post-PAR simulation can be attributed to the fact that the simulation was run with worst case operational data (85°C and 0.950V). But also actual measurements obtained at room temperature can be up to 16% faster than datasheet values for the same operational conditions. An even greater difference has also been found in [6], but such large differences (66% of the theoretical value) might be due to optimisation issues.

B. Experimental Results

The four-sensor network was implemented on all 36 devices, each sensor being composed of a 20-stage delay chain and a 1280 cycle loop counter. These parameters provide a good compromise between total sensor area, and area dedicated to delay generation. A total of 400 measurements per device were taken (100 per sensor), in order to filter statistical glitches. Fig. 3 shows the histogram of devices against their relative difference with the overall measured average (expressed as 0%). All devices were found to be faster than the expected measurement according to vendor data, with an average improvement of 13.9%. Even though the number of samples is not large, let us comment on the results. The histogram does not present a uniform or Normal distribution, which can be attributed to the fact that, even if the boards were from the same manufacturer

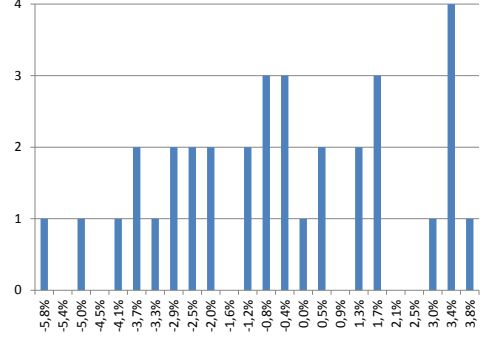


Fig. 3. Basys 2 Device Histogram. 0% Corresponds to the Measured Average.

TABLE II
INTRA-DIE VARIATION ON SPARTAN 3E.

Measure	Average (ns)	Diff. with Avg.	% Diff.
Device Overall	119947.0	0	0
Sensor ID0	119640.9	-306	-0.26
Sensor ID1	120234.4	287	0.24
Sensor ID2	119477.7	-469	-0.39
Sensor ID3	120419.3	472	0.39

batch, that is not the case for the FPGAs they mount. A certain degree of clustering and even of overlapped clusters of devices is hinted by the distribution of bars.

A closer look at the data reveals another interesting aspect of the devices under measurement. A certain amount of intra-die variation is observed within these very small FPGAs even just employing four sensors. As seen in table II, sensors 0 and 2 (rightmost quadrants) are found to be slightly but consistently faster than the global average, while sensors 1 and 3 are at the other end of the scale. To interpret these tiny differences, we must take into account studies on sensitivities and their causes [8], [9], [10]. According to these, there is a strong spatial correlation in the variability of gates closer than 1mm within a 90nm device. Given the small size of the tested devices, all intra-die variations could be attributed to random causes.

IV. INTRA DIE VARIATIONS

To further demonstrate the versatility and robustness of our sensing infrastructure, we have extracted the variability map in a larger device. The device under test was a 65nm Virtex-5 LX50T from Xilinx, on a Digilent Genesys board. For this device, the main timing limitations are the Maximum Switching Frequency (550MHz), LUT-Latch Pair Delay (T_{ito} , 0.90ns) and LUT+DFF Setup Time (T_{DICK} , 0.49ns) [11].

The design was adapted and built for this FPGA in a very time-effective way from available building blocks and in-house automation software.

A. Test Setup

A full characterisation of a bigger device requires a much denser mesh of sensors. Such sensors must have smaller footprints and take into account spatial correlation, as already mentioned, to improve spatial resolution. Following our procedure, and taking into account previous works in analysing intra-die variability [3], [1], which puts it at around 6%, a total sensor delay of around $120\mu s$ is chosen, which can be achieved with a 4-stage delay chain (each delay-stage includes 4 LUT + Latch pairs for a theoretical delay of 3.6ns) and a counter going up to 4095. Sampling frequency was set at 100MHz, and synthesis results, theoretical, simulated and measured delays can be seen in table I.

As explained in section II-A, the larger the number of sensors in a single network, the higher the measurement error [2]. To keep errors down and limit the need to ramp up the sampling clock, a matrix of 30×10 sensors was divided into 15 networks, of 20 sensors each. With our design and sampling parameters, maximum expected variation is about 700 cycles (6% variability over theoretical delay), with a maximum error of 24 cycles per sensor, or 3% of the maximum variability. Note that the networking elements for 20 sensors make use of slices (less than 3% of the 7,200 available).

B. Experimental Results

All networks are sequentially read for a period of time (in the range of a few minutes) to ensure that any self-heating is filtered out. Fig. 4 shows the variation around the average of all measured points, while Fig. 5 shows the histogram of variation around the average. As was the case with our previous inter-die experiments, the average speed of the device at room temperature is actually 21% faster than the expected speed according to datasheet information. Most of the points in the device are in the range -4% to 4% of the average measurement, demonstrating that our simple and scalable solution for device sensing can also cover specific needs for better timing information in high-end designs. Although current commercial tools do not accept a matrix on actual device timing information, a number of methods for better place-and-route have been proposed [12], which could incorporate such measurements.

V. CONCLUSIONS AND FUTURE WORK

There is a clear need for variability analysis using inexpensive and multi-purpose sensors in CMOS technology. In the case of programmable logic, the ability to obtain variability data in a robust and simple way, can turn the issue into an opportunity to further understand timing limitations and to obtain more performance out of a device. As we have shown, such highly scalable and adaptable solutions can be tailored to cover not only one aspect of monitoring along the life cycle of a device, through the usage of simple configuration tools. Our examples have focused on the early stages of that cycle,

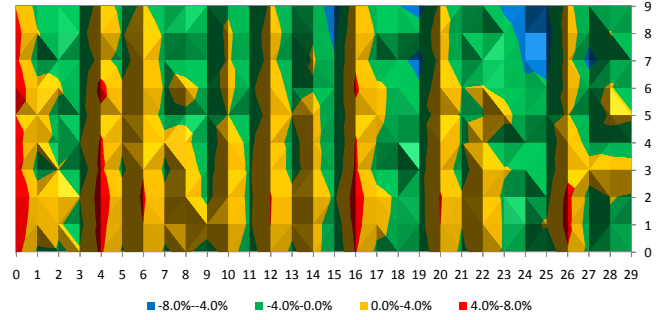


Fig. 4. Virtex-5 Sensor Matrix Difference with Device Average.

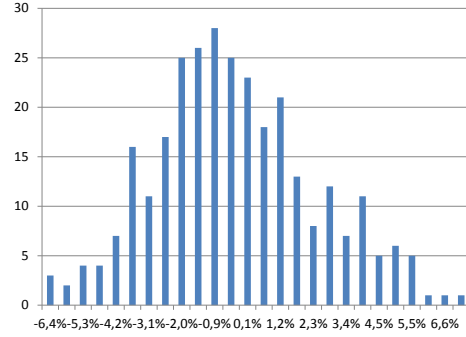


Fig. 5. Virtex-5 Sensor Histogram Difference with Device Average.

but we expect to expand its usage to cover the challenge of measuring aging variations at operational conditions.

REFERENCES

- [1] P. Sedcole, J. Wong, and P. Cheung, "Characterisation of FPGA clock variability," in *ISVLSI '08*, april 2008, pp. 322–328.
- [2] P. Ituero, M. Lopez-Vallejo, M. Marcos, and C. Osuna, "Light-weight on-chip monitoring network for dynamic adaptation and calibration," *Sensors Journal, IEEE*, vol. 12, no. 6, pp. 1736–1745, june 2012.
- [3] K. M. Zick and J. P. Hayes, "On-line sensing for healthier FPGA systems," in *Proceedings 18th ACM/SIGDA Intl. Symposium on Field programmable Gate Arrays*, ser. FPGA '10, 2010, pp. 239–248.
- [4] V. Bexiga, C. Leong, J. Semiao, I. C. Teixeira, J. P. Teixeira, M. Valdes, J. Freijedo, J. J. Rodriguez-Andina, and F. Vargas, "Performance failure prediction using built-in delay sensors in FPGAs," in *Proc. 21st Intl. Conf. Field Programmable Logic and Applications*, 2011, pp. 301–304.
- [5] Q. Chen, M. Meterellioz, and K. Roy, "A CMOS Thermal Sensor and its Applications in Temperature Adaptive Design," in *ISQED*, 2006.
- [6] P. Chen, M.-C. Shie, Z.-Y. Zheng, Z.-F. Zheng, and C.-Y. Chu, "A fully digital time-domain smart temperature sensor realized with 140 FPGA logic elements," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 54, no. 12, pp. 2661–2668, Dec. 2007.
- [7] *Spartan-3 Generation FPGA User Guide*, Xilinx Inc, June 2011.
- [8] H. Masuda, S. Ohkawa, A. Kurokawa, and M. Aoki, "Challenge: variability characterization and modeling for 65- to 90-nm processes," in *Custom Integrated Circuits Conference*, 2005, pp. 593–599.
- [9] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling within-die spatial correlation effects for process-design co-optimization," in *ISQED '05*, 2005, pp. 516–521.
- [10] Y. Cao, P. Gupta, A. Kahng, D. Sylvester, and J. Yang, "Design sensitivities to variability: extrapolations and assessments in nanometer vlsi," in *15th Intl. ASIC/SOC Conference*, sept. 2002, pp. 411 – 415.
- [11] *Virtex-5 FPGA Data Sheet*, Xilinx Inc, May 2010.
- [12] P. Sedcole and P. Y. K. Cheung, "Parametric yield in FPGAs due to within-die delay variations: a quantitative analysis," in *Proc. 15th Intl. Symp. on Field Programmable Gate Arrays*, 2007, pp. 178–187.